

Das „Dateisystem“ ZFS im Detail

Florian Hettenbach

Organisatorisches

- **Dauer ca. 1,5 Stunden**
- **Webcast wird aufgezeichnet**
- **Fragen während des Webcasts?**

Vorstellung



- **Seit 2007 bei Thomas Krenn**
- **Seit 2009 ZFS Know-how**
- **Produktmanagement Storage und Virtualisierung**
- **Business Development – Neue Geschäftsbereiche und Technologien**

Entwicklungsstand ZFS

ORACLE®
SOLARIS

 Joyent®

uvm.

DEY
STORAGE SYSTEMS

 OmniTI



 **nexenta**
Enterprise class storage for everyone

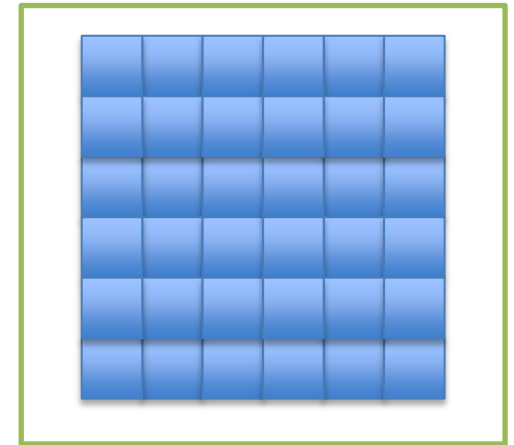
18.09.13 – ZFS Entwickler schließen sich unter Open ZFS Gruppe zusammen

http://open-zfs.org/wiki/Main_Page

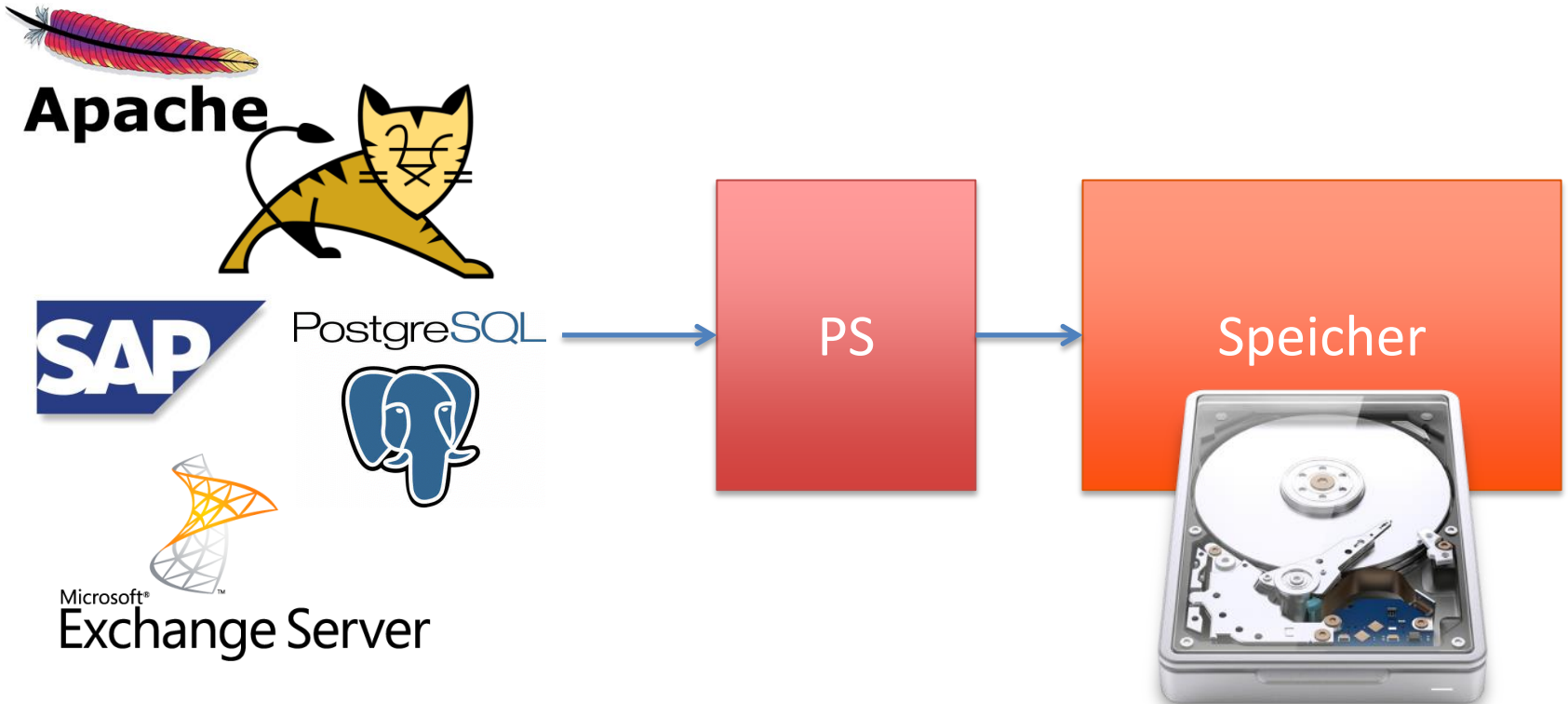
Grundlagen

Definitionen

- **Block** 
 - Speicherbereich mit fester Größe
- **Datei** 
 - Ansammlung von belegten Blöcke
- **Volume / LUN**
 - Logischer, blockbasierter „Bereich“
- **Raid**
 - Ein Mechanismus, der zur Erhöhung der Ausfallsicherheit dient – Kein Backup



Pufferspeicher



Geschwindigkeitsklassen



RAM



SSD



HDD

Geschwindigkeit

- Transferrate / Datendurchsatz
 - MB/s
 - Vergleich: Personen/h auf einer Strecke

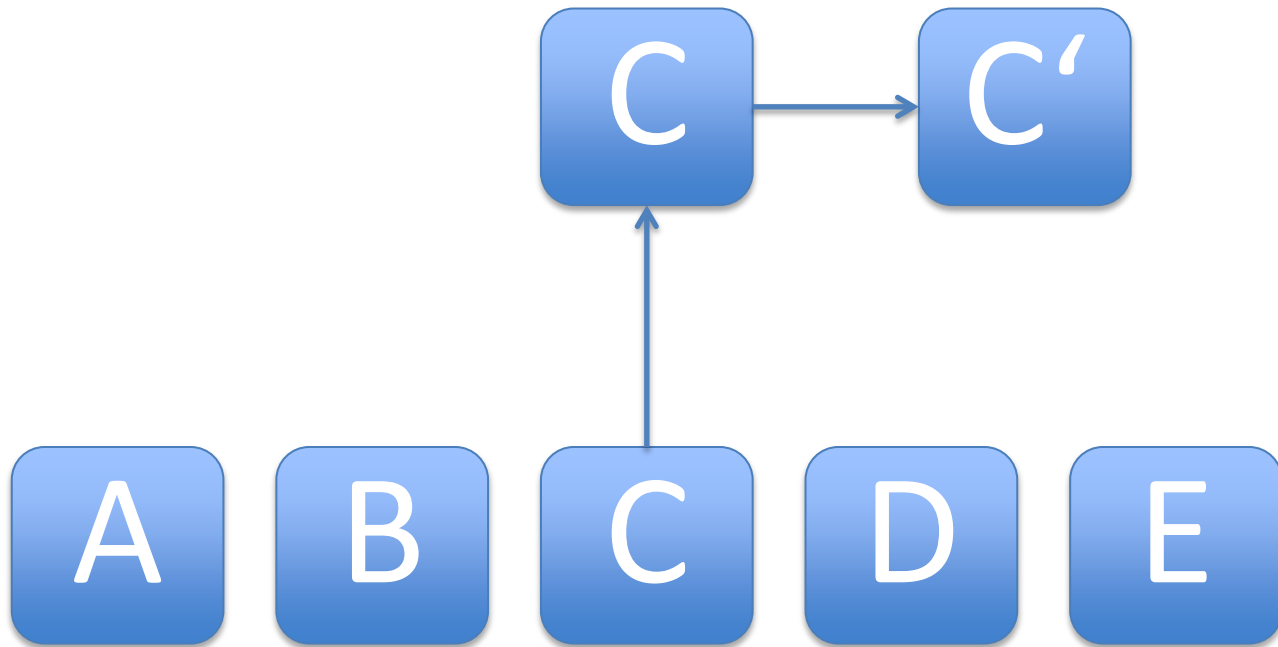
Fileserver
- I/O Operationen pro Sekunde
 - IOPS
 - Vergleich: Anzahl mögl. individueller Fahrten

Datenbanken



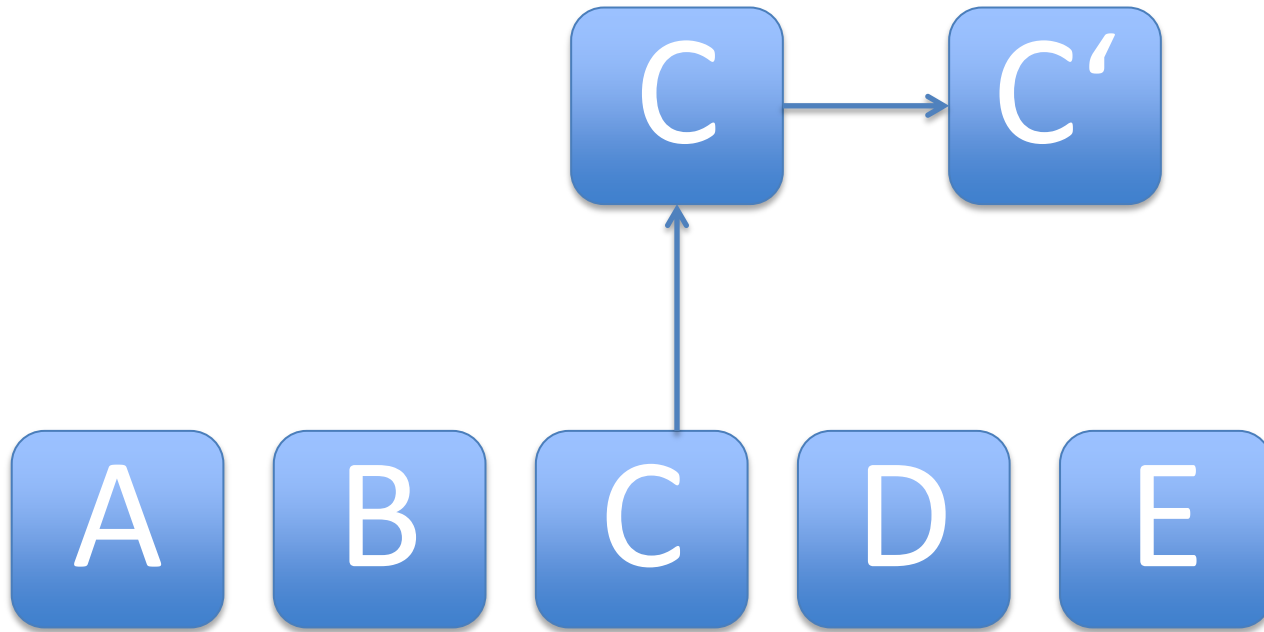
Schreibverfahren

Read Modify Write



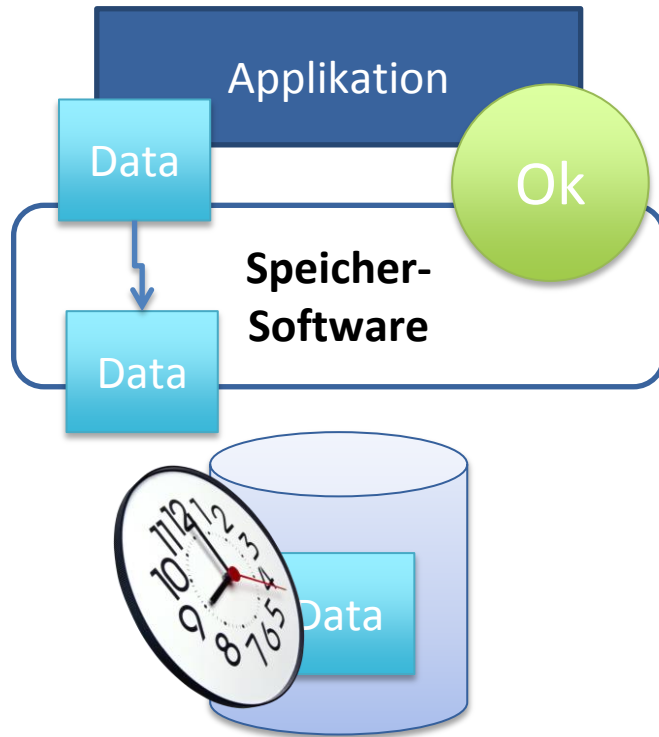
Ändern des Blocks C in C'

Copy On Write

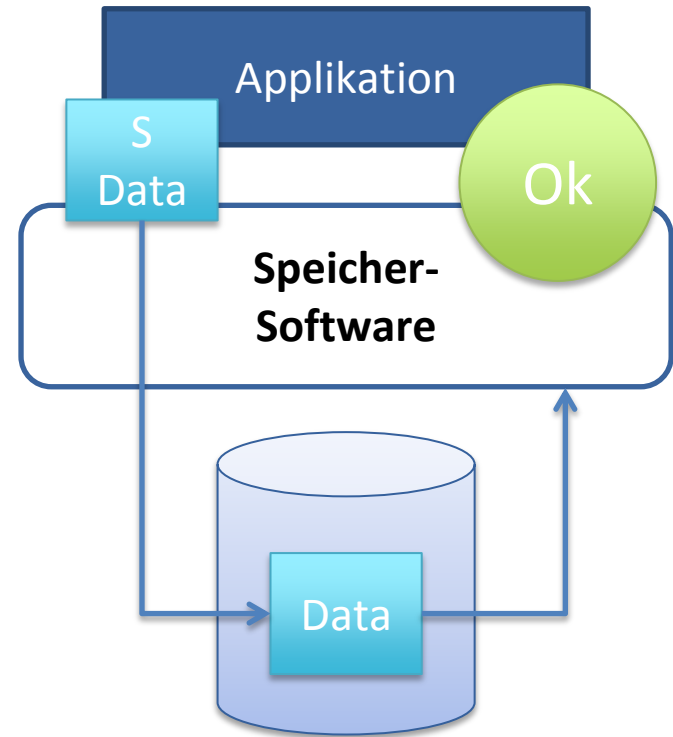


Ändern des Blocks C in C'

Asynchrone und Synchron Operationen



Asynchron



Synchron

Raid 5 Write Hole



Ändern des Blocks A (RMW)

Prüfsummen



Dokument.docx

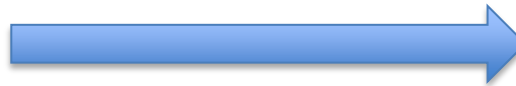


2a06b59b3e36c4799
44c88a04f8f47d25f91
4777f992adb5df15f88
f7d6d0fb4

SHA-256

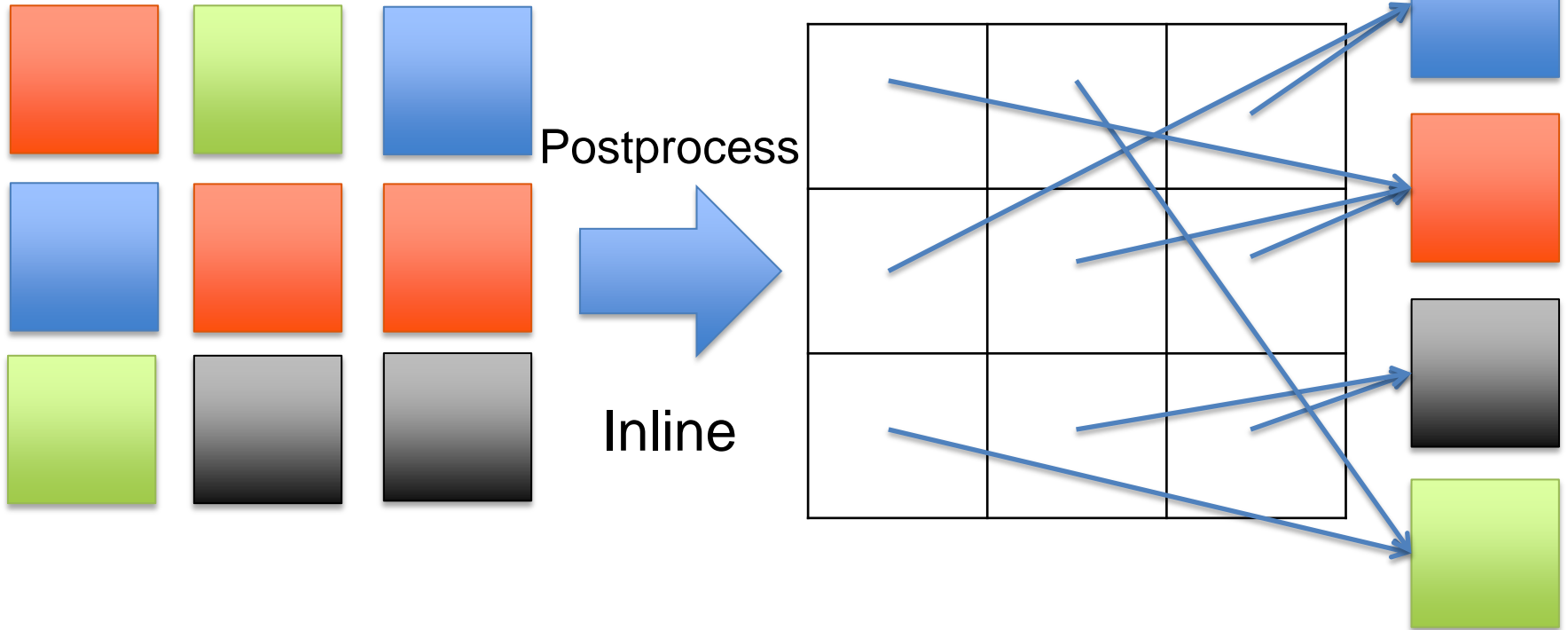


Dok1.docx



C15e77030758f2db8
8c63993a4dbe42aa4
e2c46c3f3bc3114096
4369c39e5

Deduplizierung



























Datenbankprinzipien

- **Transaktionen**
- **ACID**
 - **Ganz oder garnicht (Atomic)**
 - **Zustand muss immer K(C)onsistent sein**
 - **Keine parallelen Operationen (Isolated)**
 - **(D)auerhafte, sichere Speicherung**
- **Log**

Snapshots

Abbild des (Datei-)Systems zu einem bestimmten Zeitpunkt

	Logische Sicht	Snapshot	Belegung
12:00	   	   	   
13:00	   	   	   

Schichtmodell eines Speicher-Systems

NTFS

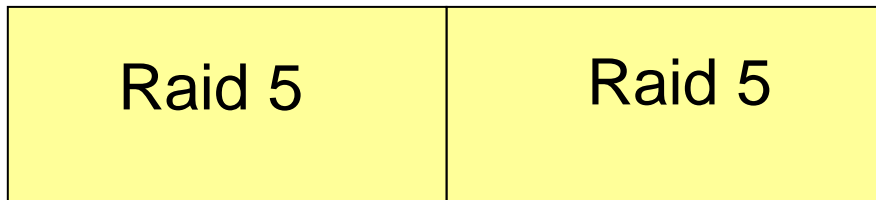
ext3

UFS

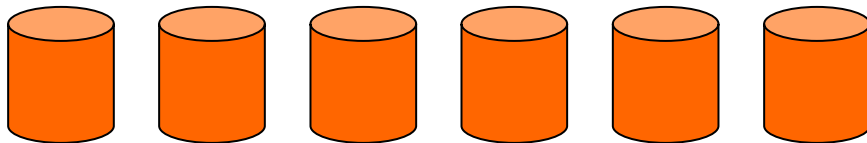
File systems



Volume Manager
(LVM)



Raid Controller



Disks



ZFS

Wo findet man derzeit ZFS

- **FreeBSD**
- **NexentaStor**
- **OpenIndiana**
- **OmniOS**
- **ZFSonLinux**
- **Solaris**

Daten Integrität

**Einfache
Administration**

**Exzellente
Geschwindigkeit**

**Enorme
Skalierbarkeit**

**Design
Prinzipien
ZFS**

```
graph TD; A[Daten Integrität] --> C((Design Prinzipien ZFS)); B[Einfache Administration] --> C; D[Exzellente Geschwindigkeit] --> C; E[Enorme Skalierbarkeit] --> C;
```

Skalierbarkeit durch 128 Bit

- **16 EiB darf ein Dateisystem groß sein**
- **2^{64} Dateisysteme in einem Speicher-Pool**
- **2^{64} Snapshots in einem Dateisystem**

Maximale Speicherkapazität:

256 Billionen Zettabytes

Kernbausteine ZFS

- **Merkle Baum**
- **End-2-End Prüfsummen**
- **Transaktionen**
- **Hybrider Speicherpool**
- **Größtmögliche Sicherheit**

Bedienung von ZFS

zpool



Pool

zfs



Dateisysteme

```
$ zpool create siege02 mirror c0t4d0 c0t6d0
```

```
$ zpool add siege02 mirror c0t5d0 c0t7d0
```

```
$ zpool status siege02
```

```
pool: siege02
```

```
state: ONLINE
```

```
scan: none requested
```

```
config:
```

NAME	STATE	READ	WRITE	CKSUM
siege02	ONLINE	0	0	0
mirror-0	ONLINE	0	0	0
c0t4d0	ONLINE	0	0	0
c0t6d0	ONLINE	0	0	0
mirror-1	ONLINE	0	0	0
c0t5d0	ONLINE	0	0	0
c0t7d0	ONLINE	0	0	0

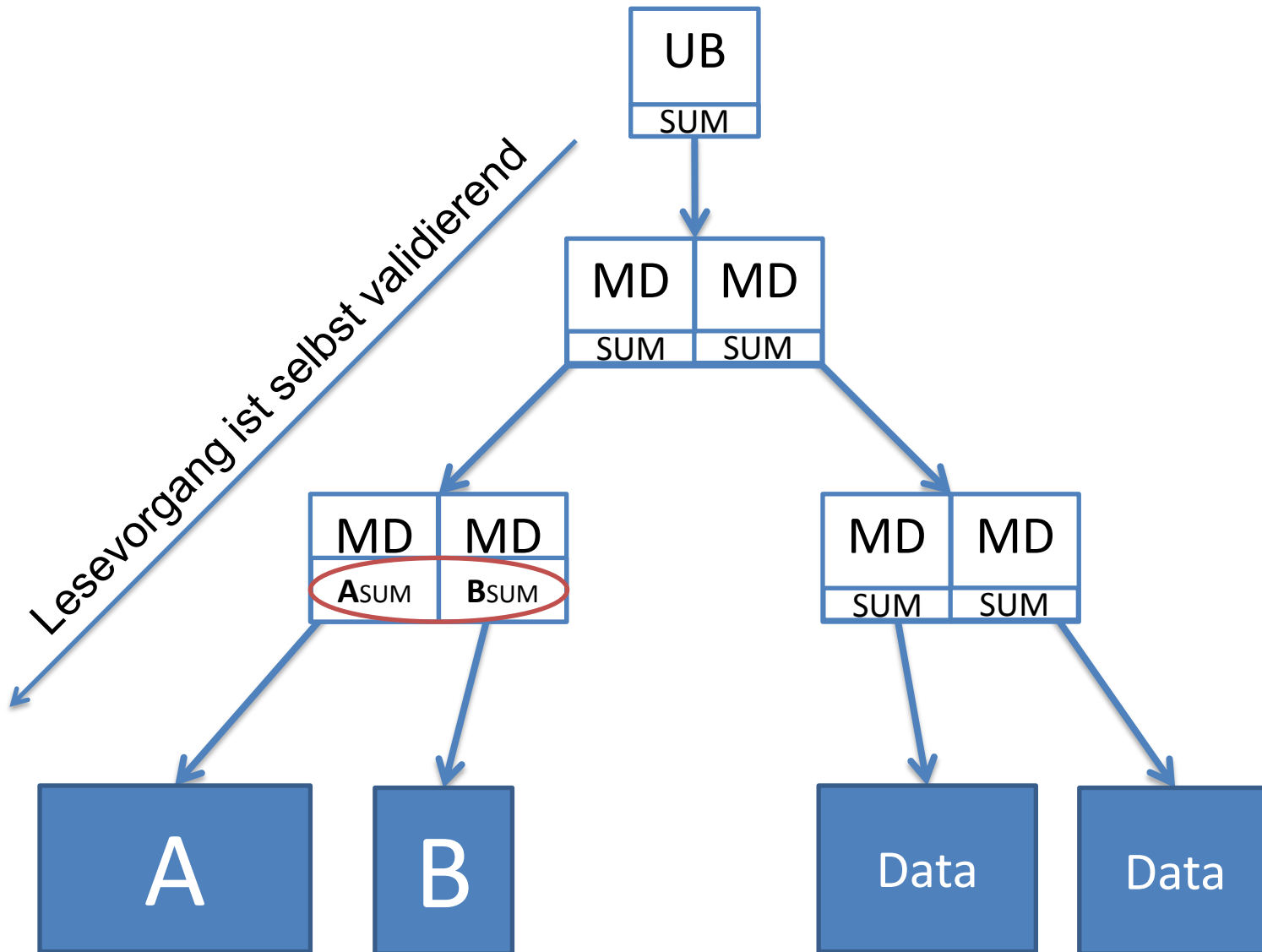
```
errors: No known data errors
```

\$ zfs create siege02/newFS

\$ zfs get all siege02/newFS

NAME	PROPERTY	VALUE	SOURCE
siege02/newFS	type	filesystem	-
siege02/newFS	creation	Sun Jun 23 22:43 2013	-
siege02/newFS	used	31K	-
siege02/newFS	available	9.78G	-
siege02/newFS	referenced	31K	-
siege02/newFS	compressratio	1.00x	-
siege02/newFS	mounted	yes	-
siege02/newFS	quota	none	default
siege02/newFS	reservation	none	default
siege02/newFS	recordsize	128K	default
siege02/newFS	mountpoint	/volumes/siege02/nelt	
siege02/newFS	compression	on	inherited
siege02/newFS	refquota	none	default
siege02/newFS	dedup	off	default

Merkle Baum



Schichtmodell eines Speicher-Systems

NTFS

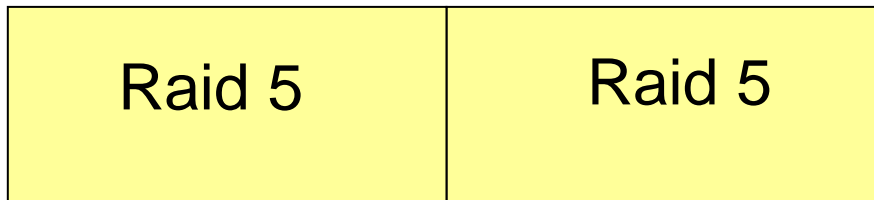
ext3

UFS

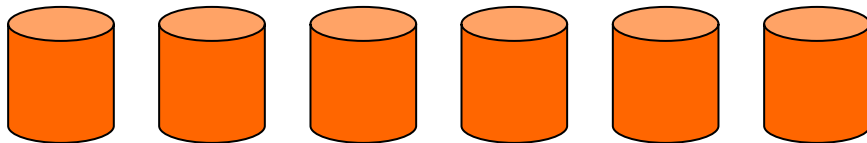
File systems



Volume Manager
(LVM)

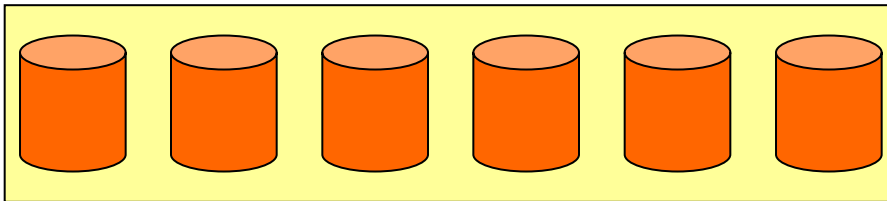
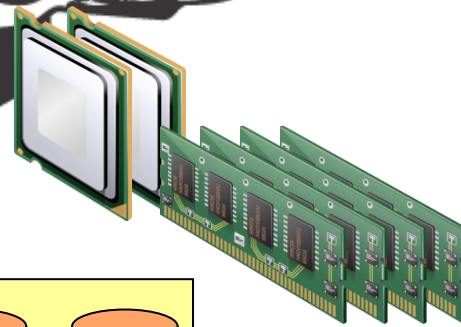
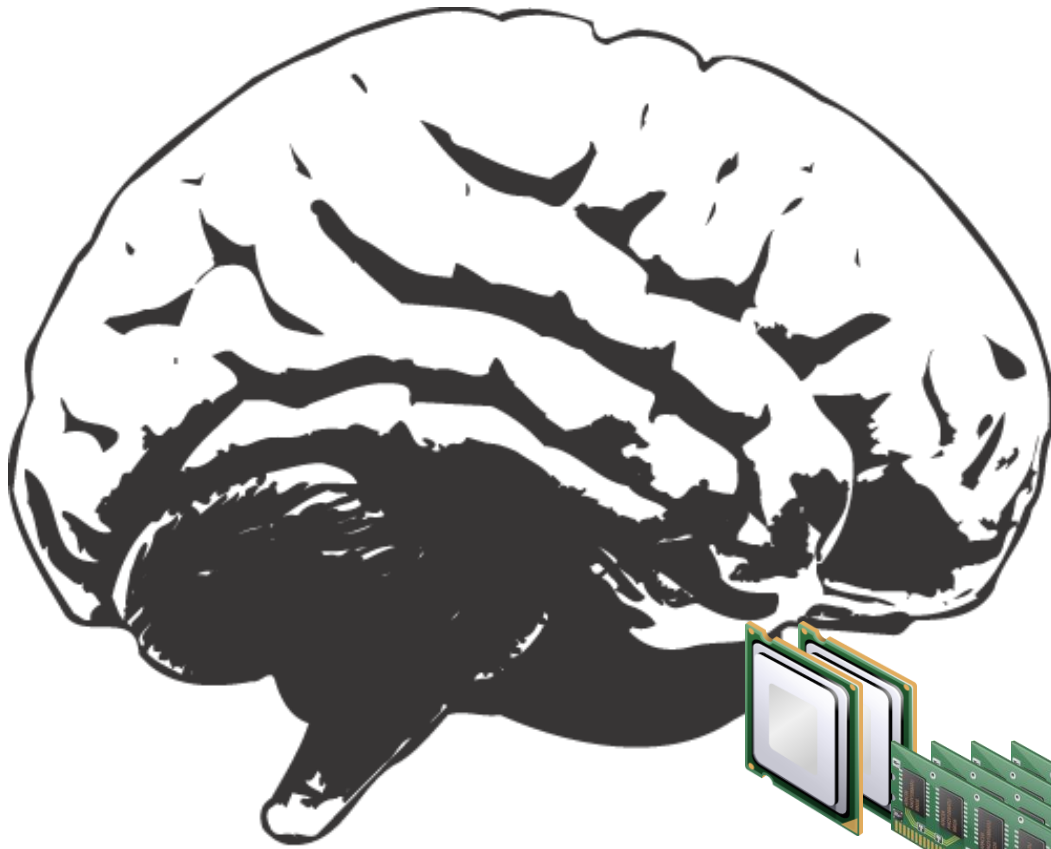


Raid Controller



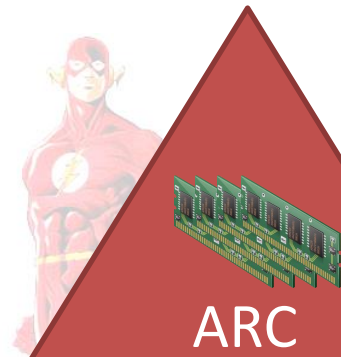
Disks





ZFS

Der hybride Speicherpool



R+W Caching

ARC



Erweitertes R+W
Caching

L2ARC

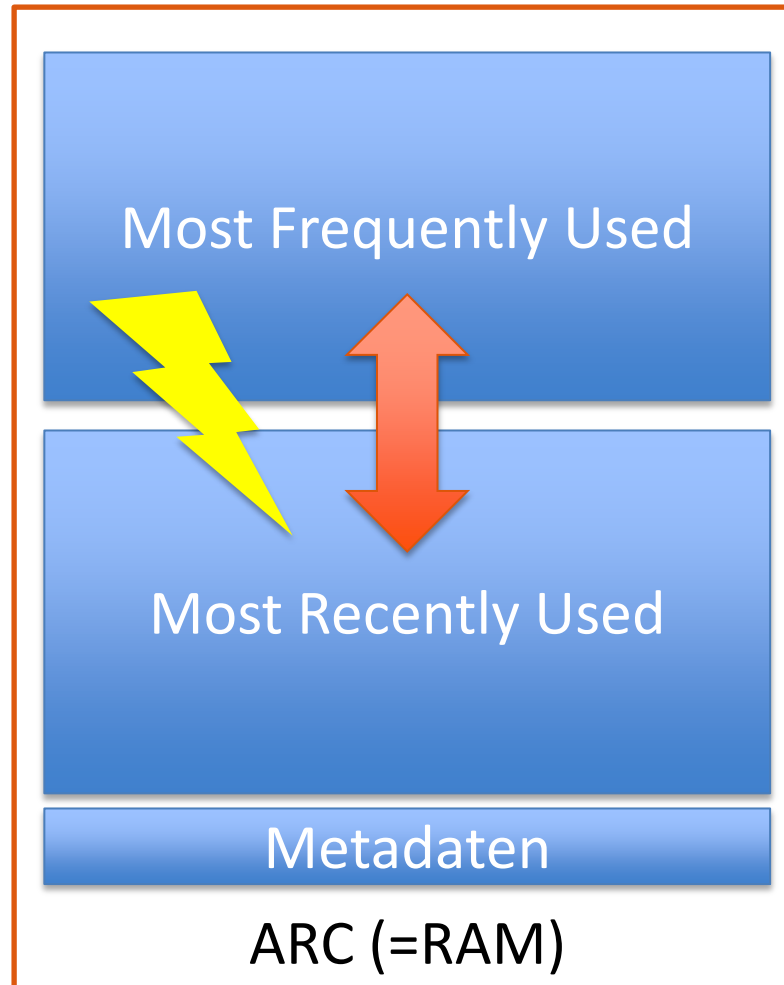
sLog



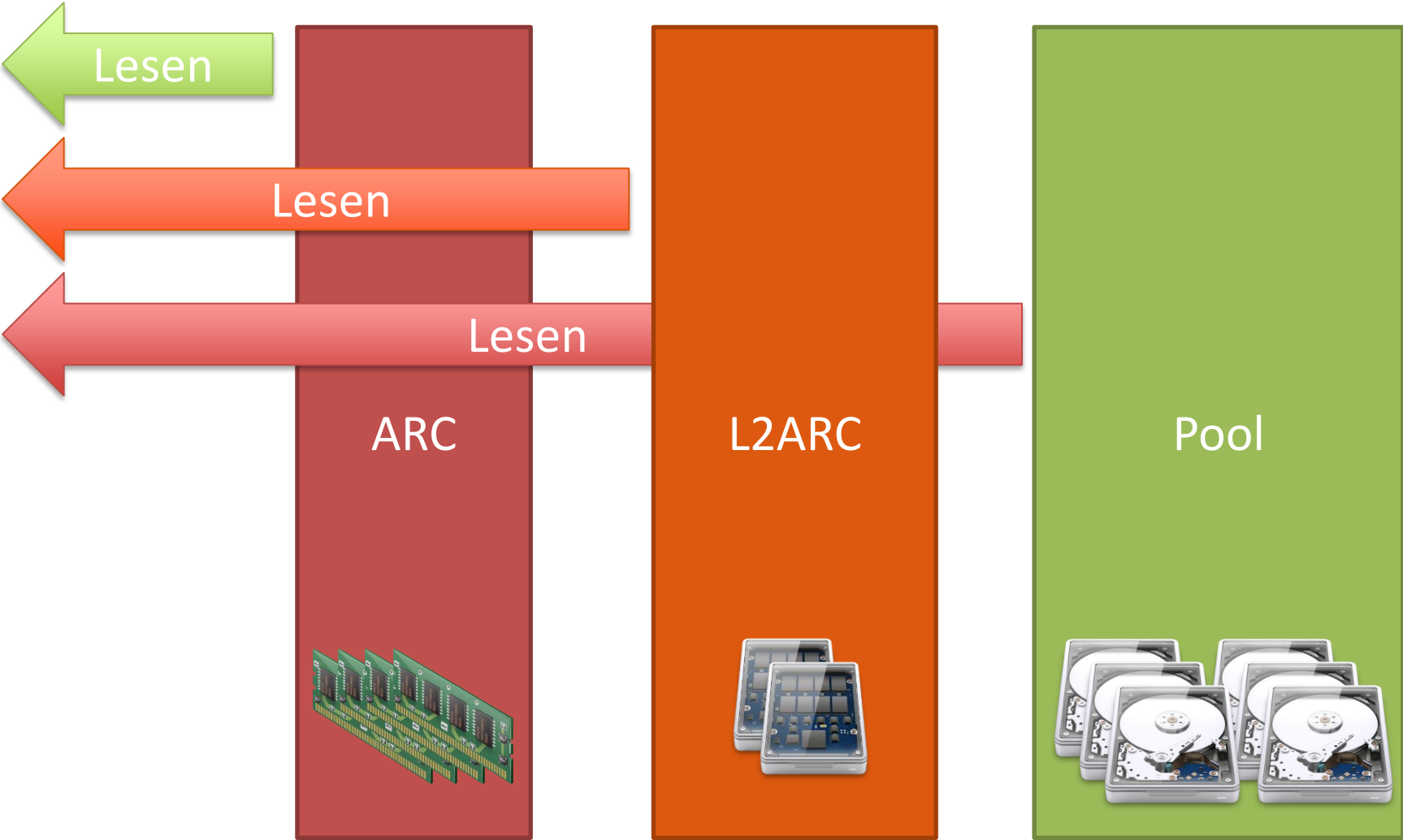
Daten

Speicher Pool

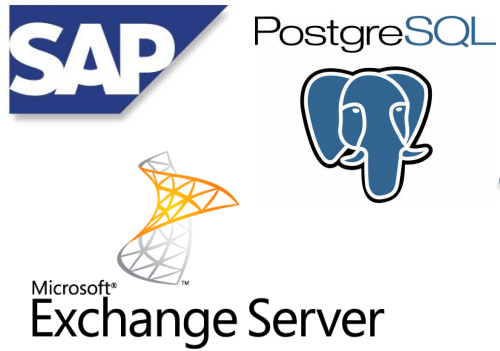
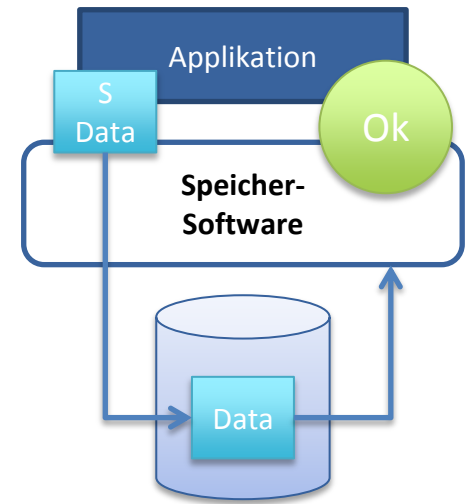
Adaptive Replacement Cache



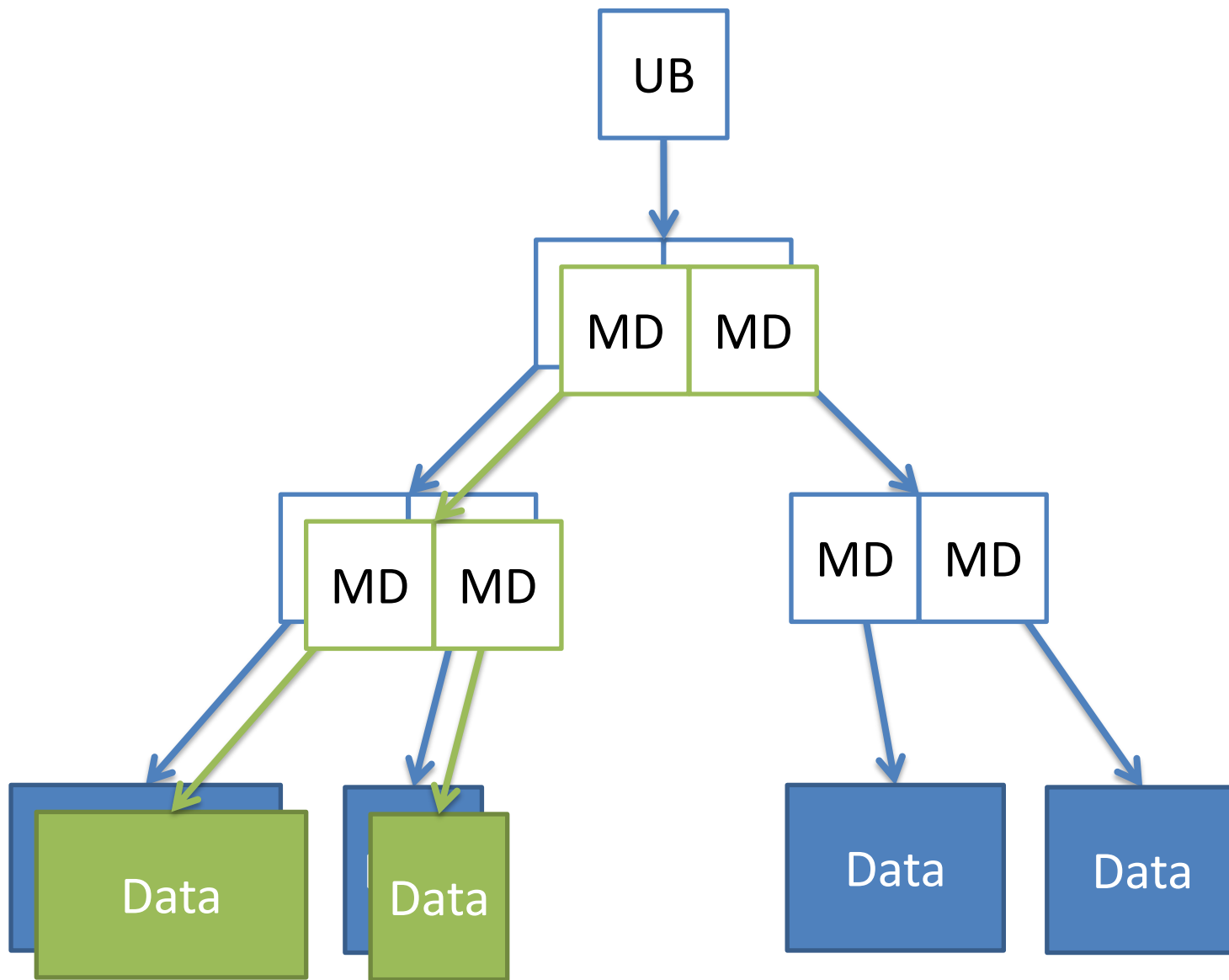
Level 2 ARC



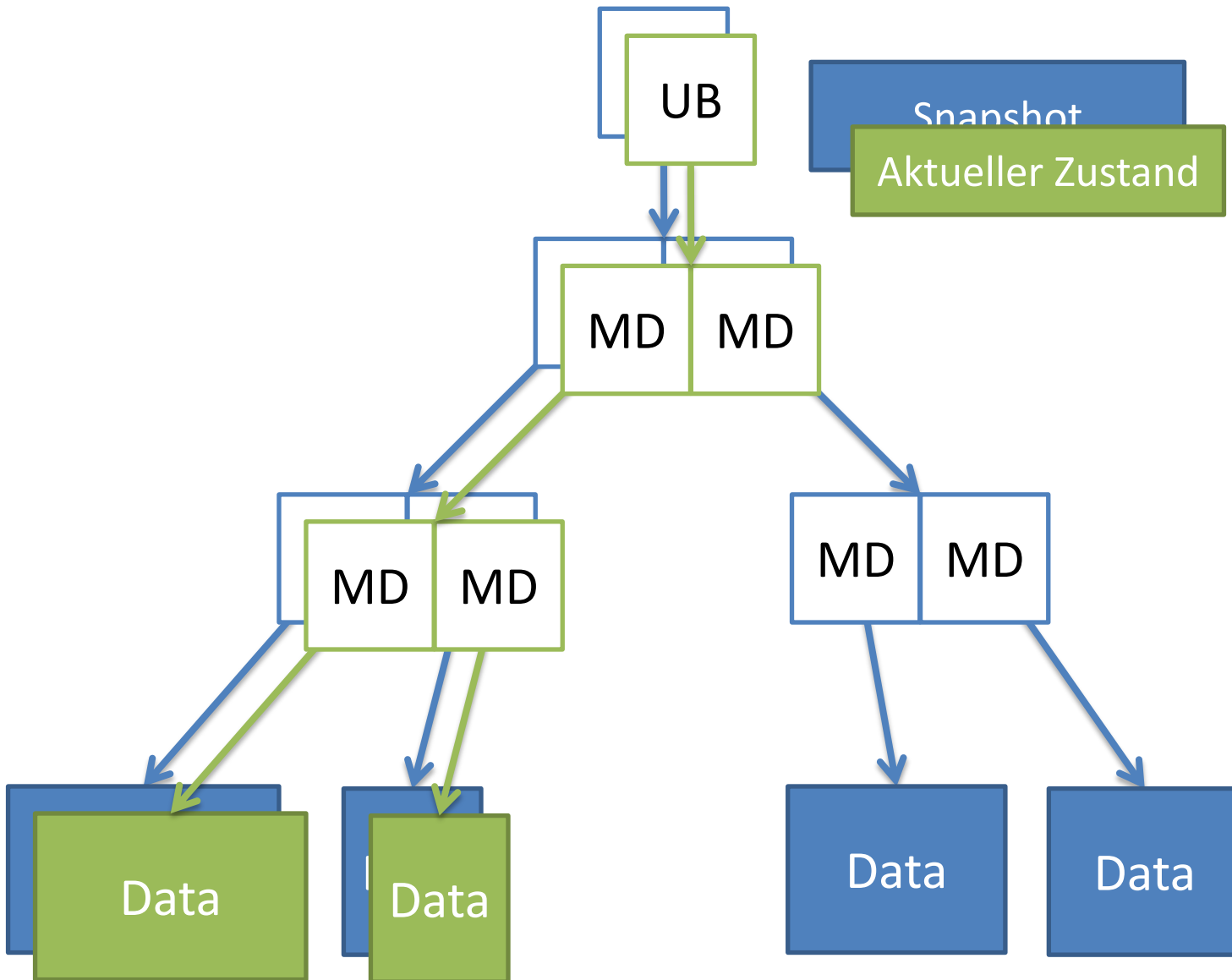
ZFS Intent Log



CoW mit ZFS

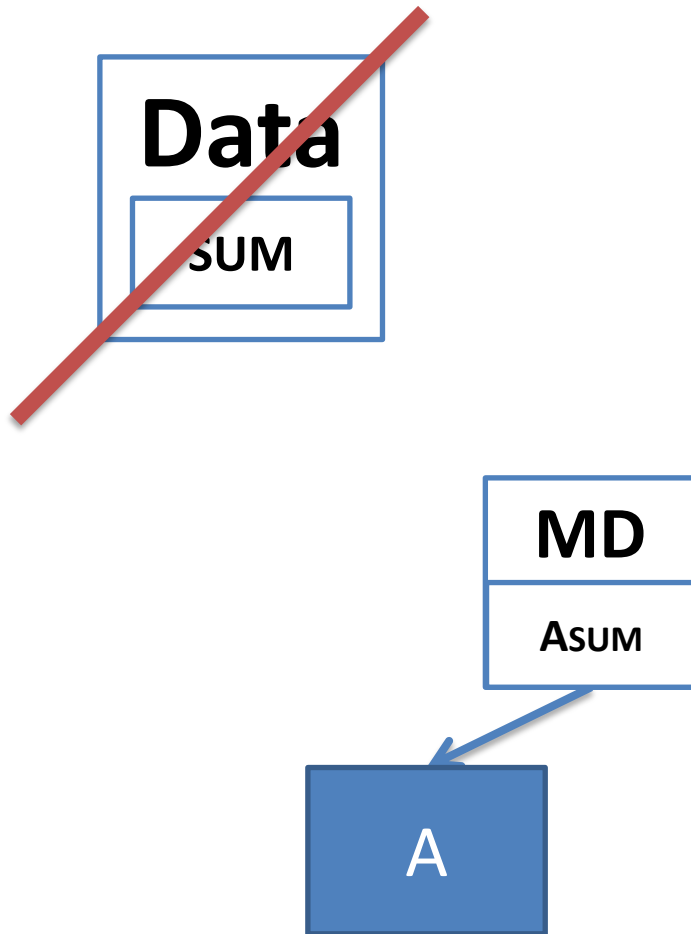


ZFS Snapshot



Sicherheit

Durchgängige Daten-Integrität



- Schutz vor
 - Phantom Schreibvorgängen
 - Treiber Bugs
 - Falschberechnete Schreib- und Lesevorgänge
 - Versehentliches Überschreiben

„Raid“-Gruppen bei ZFS

Mirror



N-1

N-2

RaidZ



N-1

RaidZ-2



N-2

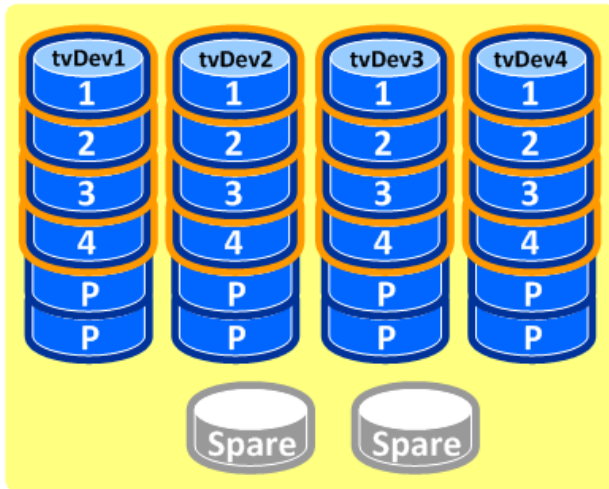
RaidZ-3



N-3

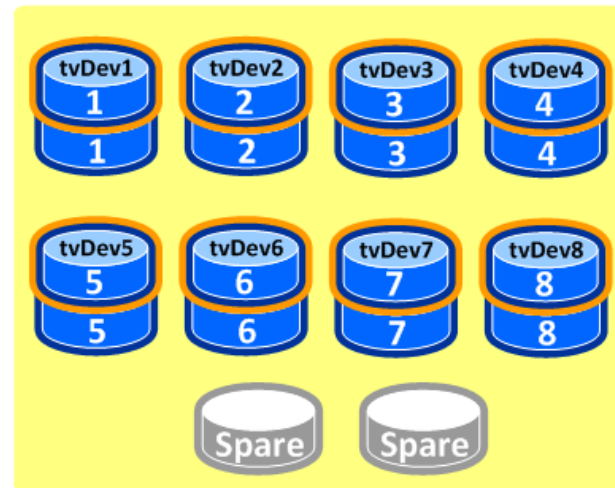
zPool

4 tvDev's (Z-2 6 disks) + 2x hot spare

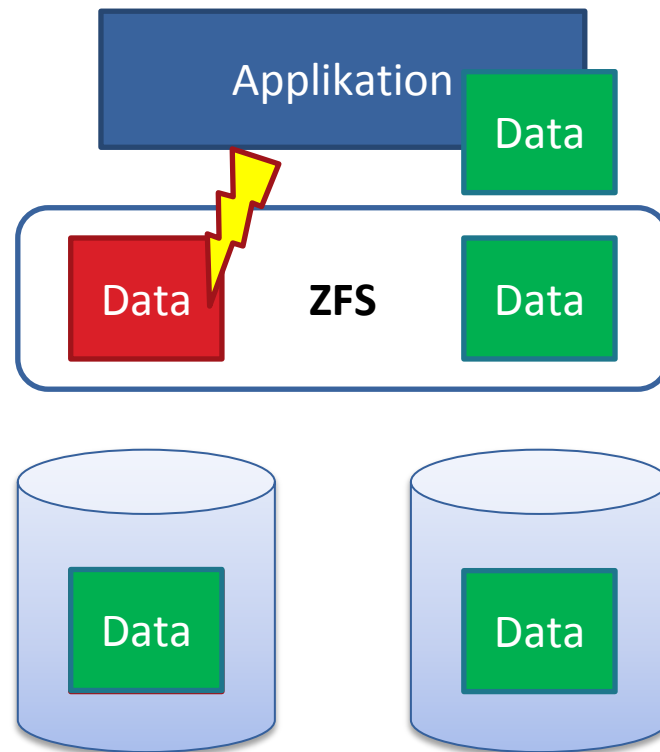


zPool

8x tvDev's (Mirror) + 2x hot spare

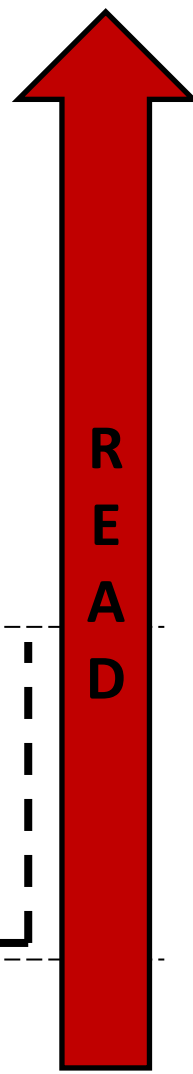
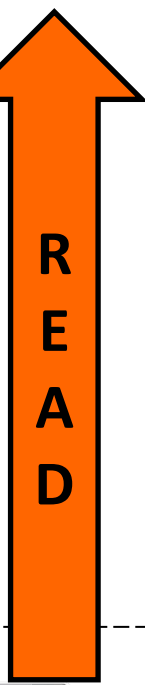
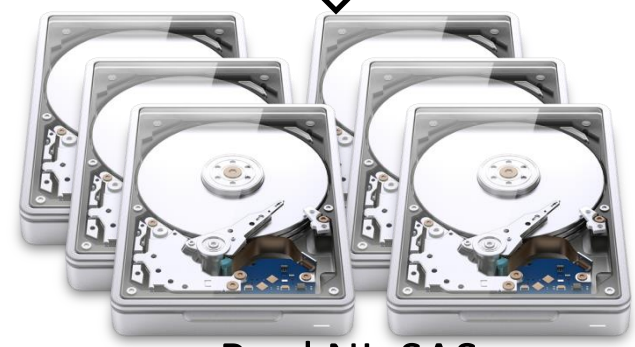
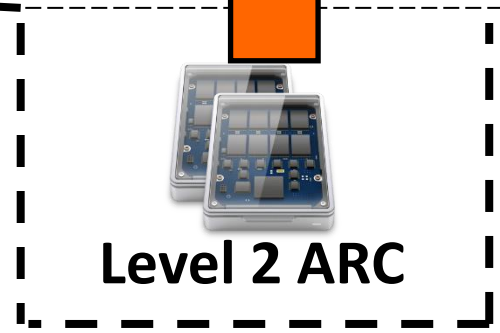
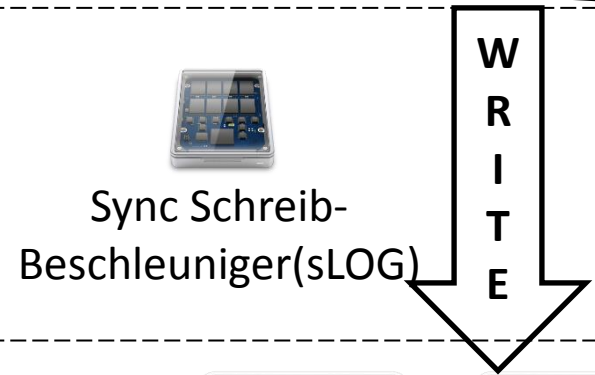
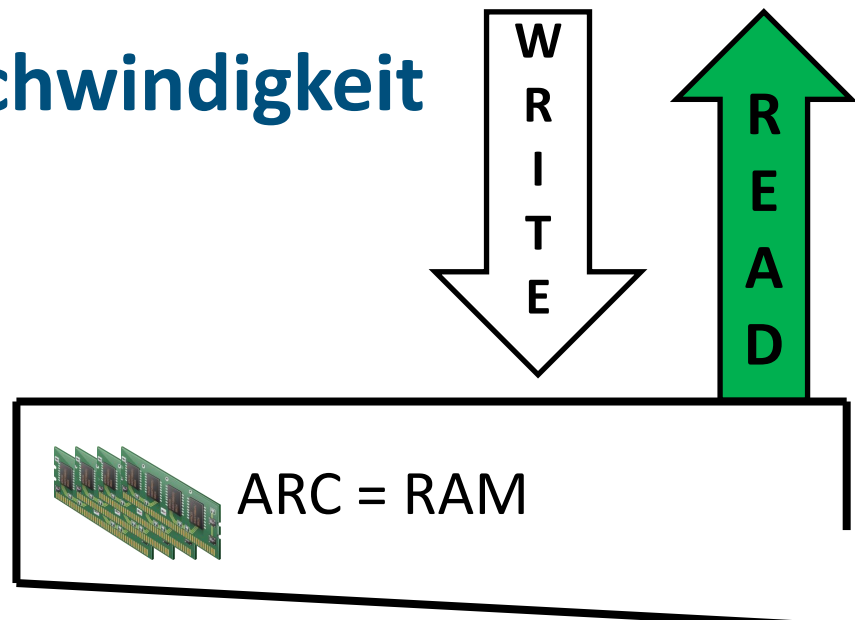


Selbstheilung in ZFS



Geschwindigkeit

Lesen- und Schreibgeschwindigkeit



Pool NL-SAS

sLOG=separates Logdevice

Praxisvorteile

Wie lange dauert ein Rebuild?



3TB



3TB

Wie lange dauert ein Rebuild?*



3TB

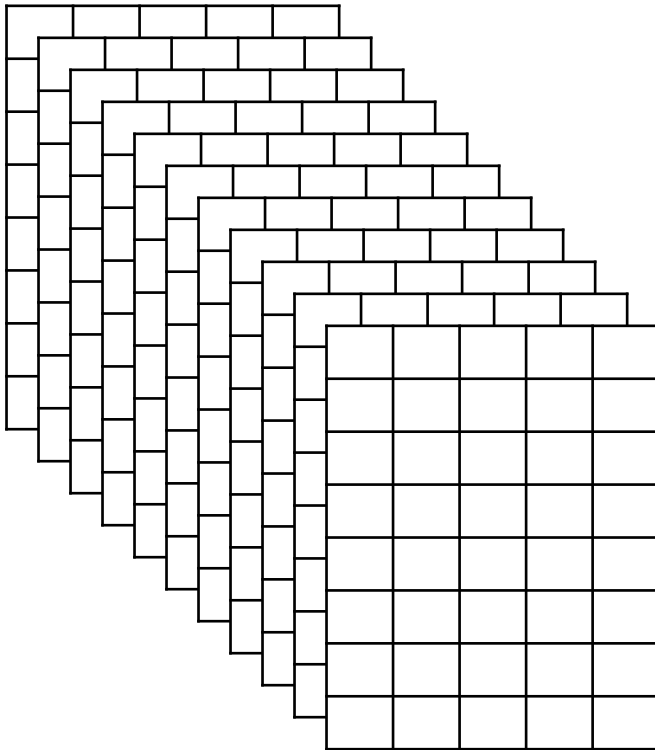


3TB

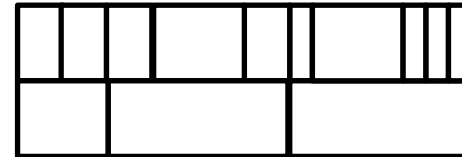
***ZFS**

Initialisierung

Raid-Controller






ZFS



Snapshot Sichtbarkeit

Netzwerk ▶ 192.168.1.105 ▶ sarge_cifsfreigabe ▶ .zfs ▶ snapshot ▶

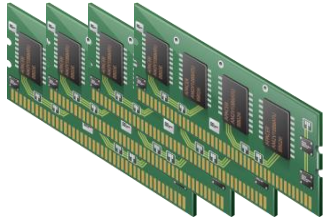
Brennen Neuer Ordner

Name	Änderungsdatum	Typ	Größe
 snap-minute-30-2013-06-23-140012	23.06.2013 13:47	Dateiordner	
 snap-minute-30-2013-06-23-143004	23.06.2013 14:21	Dateiordner	
 snap-minute-30-latest	23.06.2013 14:21	Dateiordner	

Hardware- und Pooldesign

Pool design

ARC



192GB

Zusätzliches Caching



sLog

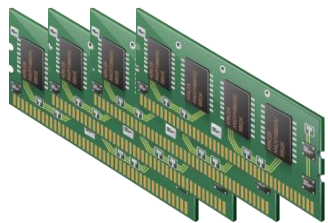


400GB
L2ARC

Pool



8x Mirror (je 2x 1TB NL-SAS)



96GB



6x Mirror (je 2x 1TB NL-SAS)



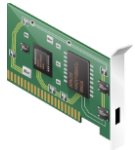
- **Geschwindigkeit:** abhängig von Anzahl der Toplevel-vDevs / Raid-Gruppen
- **Erweiterung** mit gleicher Anzahl und Typ der tvDevs

Hardwarebausteine

Kopf + JBOD(s)



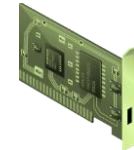
Kopf inkl. Platten
+ JBOD(s)



1GbE Karten

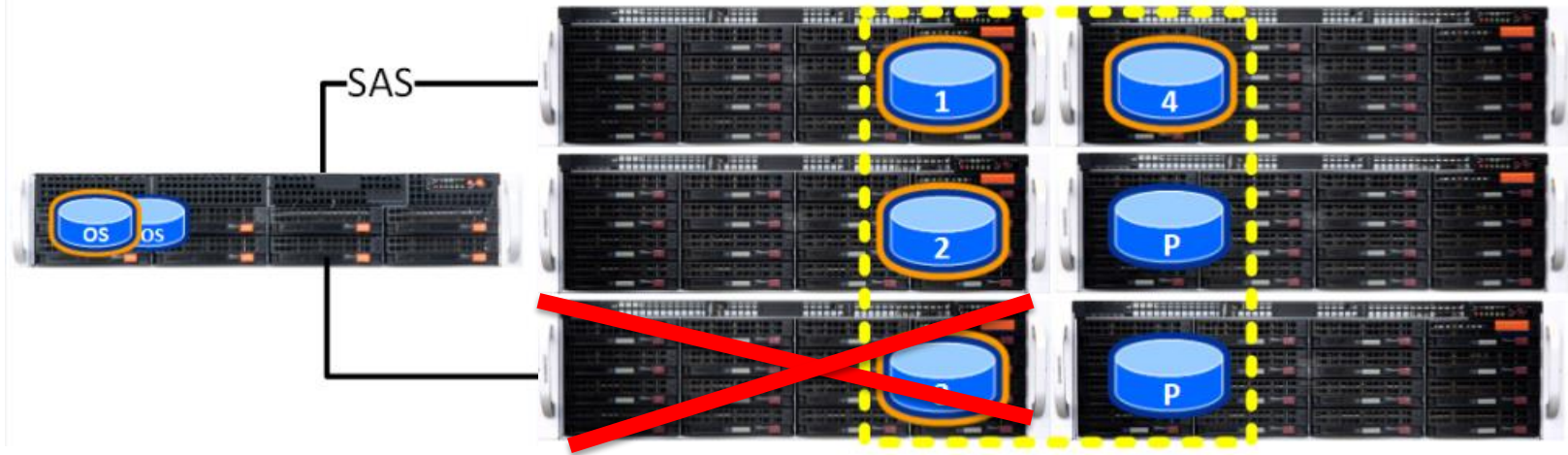


10GbE Karten

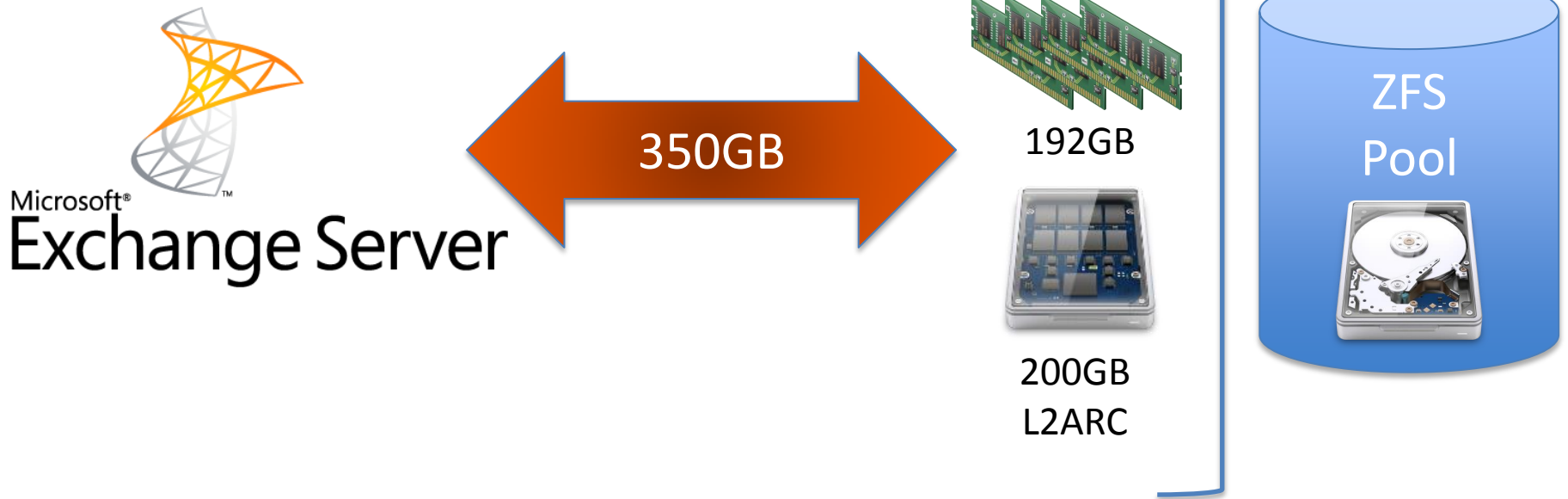


SAS HBAs

Intelligente Architektur

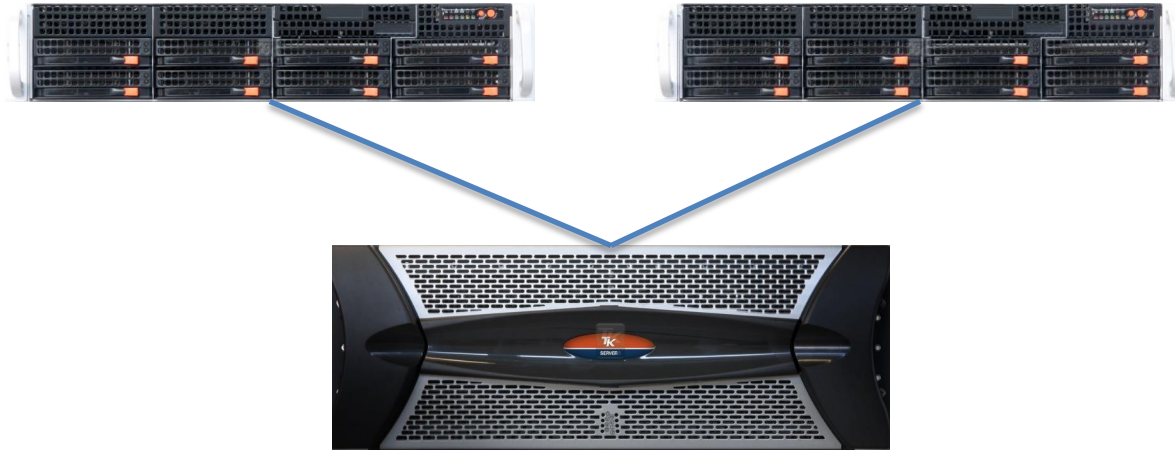


Architektur nach Arbeitslast



Zugriffe werden aus RAM und SSDs bedient!

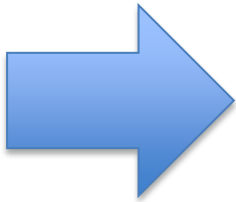
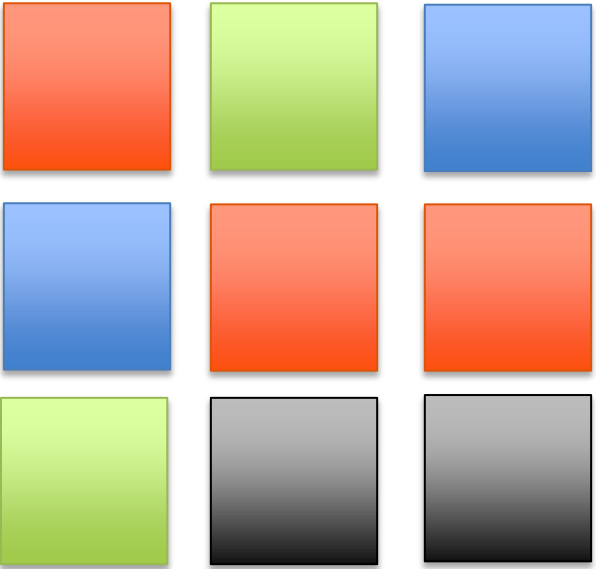
...und Hochverfügbarkeit?



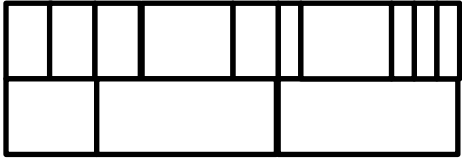
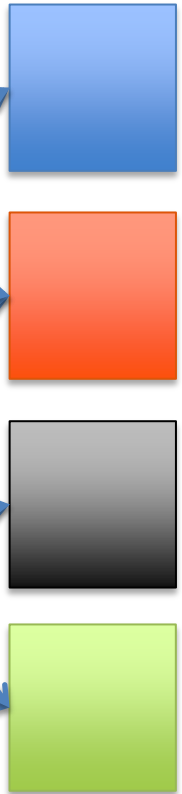
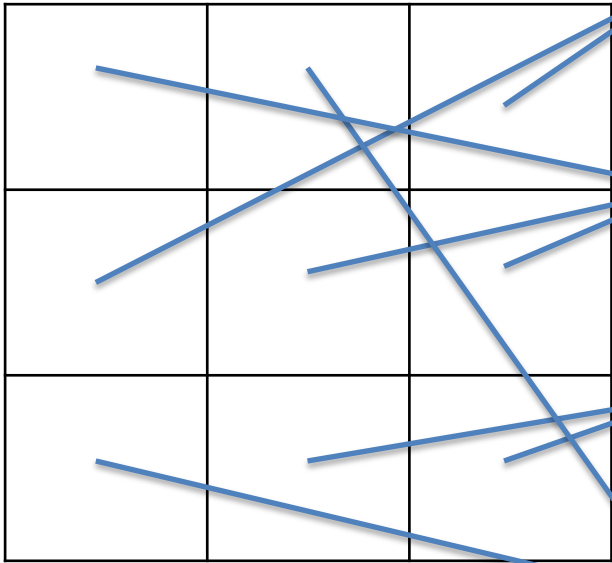
← zfs send / receive →



Deduplizierung mit ZFS?



Inline



Was es zu beachten gilt

- **RAM, RAM und nochmals RAM**
- **Pooldesign nach Anforderung maßschneidern**
- **Deduplizierung sollte wohl überlegt sein**
- **Nur SAS verwenden**
- **Link – 1st Read**
<http://nex7.blogspot.de/2013/03/readme1st.html>

ZFS Kurz und Knapp

- **End-2-End Daten-Integrität**
- **Skalierbarkeit durch intelligentes logisches und physisches Design**
- **Extrem hohe Flexibilität**

Fragen?